DOCUMENT RESUME

ED 160 081 IR 006 240

AUTHOR Surkan, Alvin J.; Evans, Richard M.
TITLE An Interactive Testing/Teaching System for Microcomputers.
PUB DATE 30 Nov 77
NOTE 15p.; Paper presented at the Annual Meeting of the Association for the Development of Computer Based Instructional Systems (Dallas, Texas, March 1-4, 1978); For related document, see IR 006 231

EDRS PRICE MF-$0.83 Plus Postage. HC Not Available from EDRS.
DESCRIPTORS *Computer Assisted Instruction; *Computer Programs; Cost Effectiveness; *Instructional Systems; Man Machine Systems; Mastery Learning; Pacing; *Testing
IDENTIFIERS APL Programing Language; *Minicomputers

ABSTRACT

Results of the experience gained in designing and using an autonomous computer-assisted testing system are reported, including conclusions based on more thar two years testing with typical classes. In this system, definitely expandable auxiliary storage for testing items and programs or scores is provided. Cathode ray tube display of questions and reinforcement, and real-time checking of answers is followed by recording partial scores and updating the number of trials by each student on each exam. The system provides a self-contained, portable, and economical facility well suited to variable-paced, proficiency-based courses of study. (Author/VT)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED160081

U S DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

An Interactive Testing/Teaching System for Microcomputers

Alvin J. Surkan

Department of Computer Science

University of Nebraska

Lincoln, Nebraska 68588

and

Richard M. Evans

Department of Education

Morningside College

Sioux City, Iowa 51106

30 November 1977

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

John J. Hirschbuhl

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

IR006240

Abstract

An autonomous computer-assisted testing system has been realized through programming a portable computer system in APL. Indefinitely expandable auxilary storage for testing items and programs or scores is provided by alternate use of a pair of removable cartridges with tapes of 204 kilobytes. A single cartridge provided an item pool of nearly 100 items. Each item consisted of a question, correct answer, and reinforcement. The second tape stored user programs plus up to 107 files, each with up to 20 sets of indices and scores for 15 randomly selected questions. Cathode ray tube display of questions and reinforcement and real-time checking of answers is followed by recording partial scores and updating the number of trials by each student on each exam. The system provides a self-contained, portable and economical facility well suited to variable-paced, proficiency-based courses of study. Results of the experience gained in designing and using the system are included with conclusions based on more than two years testing with typical classes.

An Interactive Testing/Teaching System

for Microcomputers

Introduction

Currently available implementations of high level language on portable, desk top sized computer systems equipped with significant main and auxiliary storage are rapidly expanding opportunities to create elaborate programs which control and tailor the presentation of learning items. With these new compact systems the per-student-hour cost can be five to ten times lower than that currently realized with CAI delivered by large computer systems.

This paper presents the results of experience gained while designing, implementing and putting in regular classroom use, an autonomous IBM 5100 computer system for testing, scoring, and providing reinforcement. The item pools used in the development of the system were adapted from the instructor's manual accompanying Biehler's (1974) textbook: Psychology Applied to Teaching.

The system, as currently implemented, affords a practical and economical opportunity to explore objectively, the effectiveness of computer delivered testing, and its effect on achievement in a proficiency-based mastery learning course of study that both employs partial scoring and permits time-variable pacing. Proficiency-based implies that the study for a course is directed towards the goal of attaining prescribed-in-advance levels of test score totals on a preset number of tests. These levels if achieved within a limited number of trials earn designated final letter grades. Partial scoring means that points are awarded for any correct responses that are obtained after one or more wrong answers until all alternate choices have been exhausted.

Because the mastery learning system of instruction allows students to arrange to take a test both as soon and as often as they feel prepared and also because the rate at which subsequent questions are received depends directly on how quickly the last question is answered (which is totally under the control of each student), the variable pacing automatically geared to the students' ability is an important feature of the system. This has been made practical by the economical implementation of the testing and reinforcement system on an IBM Model 5100 portable computer. The low cost of the system permits nearly unlimited access to testing and reinforcement at convenient times.

Hardware Configuration

The system used is a completely self-contained IBM 5100 portable computer configured with 32,768 eight-bit characters of main memory and a read-only memory containing an APL interpreter. The device is equipped with a single built-in tape drive and CRT video screen with 14 64-character lines of display. Auxiliary storage was provided by removeable tape cartridges, each capable of storing over 204,000 characters. One tape cartridge serves to store the system of programs and all item numbers and scores earned by the students for all trials of the tests associated with a single course. An additional tape cartridge capable of storing up to 107 test items was required for each test in a course. This hardware configuration is highly compact and portable, requiring for operation nothing more than a three ampere source of 110V ac power. Since only the video display is used, no paper or other hard-copy medium is necessary.

Identification of System Functions

The complete testing system contains three sets of primative functions.

These sets of functions are identifiable with three classes of computer users. The first class represents the instructor who creates and/or improves testing items. The second class represents the students who at the computer make trials at answering random subsets of questions on each of several tests. The third class is the educator or research worker who uses the computer to analyze the performance of students and testing items.

The necessary functions identified for each of these three classes of users are listed in the following subsections:

Instructor's Item and Score File Maintenance Functions

1. Create or improve test item files.
2. Mark, tag and initialize score files.
3. Setting of testing parameters.
4. Validation of identification numbers for admitted students.

Student's Testing Program Functions

1. Identification of Student.
2. Checking for valid admission and exceeding limit on trials.
3. Seeking and identification of question tape.
4. Introduction of student to testing objectives.
5. Generation of random test item numbers.
6. Sequencing of testing item numbers for retrieval.
7. Retrieval of components of a single test item.
8. Presentation of individual questions.
9. Limit checks on numbers of question attempts and test trials.
10. Recording of scores for each trial.
11. Updating of status trial number for each student and test.

Researcher's Item and Student Score Analysis Functions

1. Setting of constraints defining the subsets of the scores of interest.
2. Identification and selection of score subset meeting constraints.
3. Delivery of selected score subsets to statistics program.
4. Histogram data acquisition and display.

Organization of System Functions

Overview of Testing Functions

The testing system was implemented in terms of variables and functions defined in the array processing language APL (Iverson and Falkoff 1957) for which interpreters are widely available on most major computer systems and a growing number of smaller and portable systems. The names of the thirteen functions that manage the presentation of questions, answer checking, and recording of scores are APPEND, ASK, DELAY, FETCH, FORMAT, IDT, INTRO, LABEL, PAC, QUIZ, SEEK, SHOW, STATISTICS, and STUDENT.

Testing is initiated by the keyboard entry of the function name STUDENT. This function first calls the function IDT provided for identifying the tapes and students. With the repeated use of the function ASK, the IDT function obtains the date and the number of the test that is to be taken. The student's first name is also requested to enable the student-program dialog to be personalized. Beside acquiring these data, the function also evokes another function called INTRO which presents text explaining the testing system to the student. Finally, after checking that the student is not attempting to exceed the maximum number of trials permitted for the particular test specified, the function IDT either rejects the student or calls the function SEEK to seek the appropriate question tape. On succeeding to sense that the correct tape is present, SEEK returns control to the initiating function STUDENT.

A message is then presented to alert the identified and accepted student to anticipate the arrival of the first question. The DELAY function permits this message to be displayed on the screen momentarily before STUDENT finally calls the function QUIZ which then uses the following global variables as indicated:

COURSE --- number of the course file open for testing

DA --- number of the day of the month

DISPLACEMENT --- number of files before first student or item file

FN --- first name of the student being tested

IDN --- number by which the student is identified in the course

IDNOS --- 3-D bit array indicating status on admission and number of trials by test and student

MO --- number of the month in the year

NTOTAL --- number of random question in test pool

QUESTIONS --- number of questions to be randomly selected

TEST --- number of the test for the course

D --- identification number, displacement, number of files, random link

Utility Functions for Test Creation and Score or Item Analysis

The system for identifying, controlling testing, and score keeping is made generally useful with several additional supporting functions that are not needed during normal use of system in testing. For this reason, we will only outlive their general function rather than explain every instance that they are evoked for tagging new files or selectively retrieving specified subsets of the scores already stored.

ADMIT --- permits instructor to validate the identification number of any new student to be tested

ASK --- repeatedly requests entry of numerical or literal data until it conforms with controls imposed on type and number and length

FIX --- line editing function for altering lines of text within the items

FORMAT --- puts item numbers and scores in proper format for display

HEAD --- creates heading for viewing of scores

NEWCLASS --- resets all history of previous course admissions and test trials to zero

MBR --- tests if vector elements are members of intervals for selecting subsets of scores from files

PASSWORD -- produces exit from function NEWCLASS if no proper password is provided

PUTQAF --- inserts all components of a test item into a file

SELECT --- resets constraints on which subset of student scores are to be inspected

SELECTOR --- determines whether or not members of the file retrieved fit the constraints

TAGFILE --- labels all score files with course and student numbers

TEXTIN --- assists with building of regular arrays for test items

UPC --- unpacks character representation to produce a numerical form

Test Management Functions

The central function of the system, QUIZ, manages the presentation of questions, checking of answers, display of feedback or reinforcement frames and scoring. After generating a random set of numbers for choosing items, this function calls the function GETQAF which selects the file containing the correct item and checks the label on the file for the proper course and item numbers. If both numbers are right, the function fetches the three item

components QQ, AA, and FF. These variables provide the question, answer, and feedback frames, respectively. QUIZ repeats the presentation of the question up to four times until the answer provided by the student matches with the content of AA. At that time, the item number and score are placed in an array named SCORE, by the function STATISTICS. QUIZ then displays FF, the feedback frame to reinforce the students' ideas about the completed question, proceeds to the next, if any, remaining items, numbers to present sequentially again the question QQ, checks the answer AA, and finally provides FF the feedback.

Upon completion of all or more than half of a preselected number of test items, the QUIZ function requests that the tape cartridge used for storing scores replace that tape storing the items and calls the function SEEK which then confirms that indeed the correct tape cartridge has been inserted. If not, SEEK repeatedly asserts that the wrong tape has been sensed until the proper one is in place.

The function APPEND upon being provided with the student identification number, computes the proper address of that student's file and evokes the function FETCH to retrieve a packed copy of all the old scores of the student previously recorded. These old scores are then rewritten back to the same file and followed by the new scores prefixed by eight elements of identifying information. These elements are: (1) a zero for flaging the beginning of each new record; (2) the identification number assigned each student for that course; (3) a two-digit number for the year; (4) the course number; (5) number of the month; (6) number of the test in that course; (7) number of the day; and (8) a sequence number indicating on which trial, of that particular test, for which the attached scores were achieved.

Upon completion of the filing of the students' new scores, the three-

dimensional array of bits IDNOS with dimensions corresponding with information on test number, status, and student identification number is modified. Each time the two last-bit positions of the second dimension are incremented to reflect the number of times that each student has completed a trial on each particular test. This information is preserved for limiting the number of trials that a student is permitted to be tested with replicates of items for each test. Also, the first bit of the second dimension of the array IDNOS is currently used to indicate if the student number is validated to take all tests. Alternately, this layer of the array could be used to control selectively the admission of specific students to specific tests.

An auxiliary program named STATUS has been created for checking the bit array IDNOS to show what stage of testing the various students have reached.

Upon completion of score recording, QUIZ evokes the function SHOW which displays the scores earned for each test item and the percentage of the total score. The tape with the scores is then relabeled to reflect cumulative changes in a random link used for item sampling and the status of array IDNOS, indicating the status of testing for all students in the course. Before exiting from QUIZ and finally STUDENT, instructions are given to the student as to how to view his scores again or to proceed to another test or replicate trial of the current test.

Observations on Implementation Decisions

The preliminary design of the testing system used an IBM 5100 computer initially configured with only 16,384 characters of main memory. This version did not include functions required for both identification of the students and tests and the storing of scores. A single tape cartridge of auxiliary storage was used at first to store the system of programs and each item pool

of test questions, answers, and feedback frames. After gaining experience with the single tape system, the design considerations of a two-tape system with score recording were established. Assuming that the question, answers, and feedback would require storage for up to two CRT screens (each of 14 sixty-four character lines or a total of 1792 characters) and recognizing that tape files may be marked only in multiples of 1024 characters, it is observed that the optimum file size must be either 1024 or 2048 characters. Choosing mixtures of these two file sizes yields a range of 81 to 136 for the number of items available from a single cartridge. In view of the fact that the number of items in a single test tape need not in practice exceed 128 or at most 256, question numbers can be represented in storage as single characters. Also because the course, test, student, trial, year, month, and day numbers can be specified by integers in the same range, they too, can be stored as single characters. Thus if 1024 character files are established for storing sets of 15 scores each with eight characters of identifying information, one can expect to store nearly 1024/(8+2x15) or approximately 26 such sets of scores in each of 107 student files. For each student a single 1024 character file would be sufficient to store all such sets of scores from a maximum of three trials on each of up to eight tests per course.

To introduce into the system a second tape to permit the storage of scores, it was necessary to implement functions that would identify both the tapes and the students, then interactively acquire other test identifying information needed to manage the creation and modification of score files. Tentatively this necessitated expanding the main memory of the IBM 5100 from 16384 to 32768 characters.

It has been found that with proper tuning of both the input functions

for retrieving questions and the output functions that append scores after each test, it is quite practical to realize an economical testing system capable of delivering randomly selected items at a rate of 30 to 45 items per hour.

Conclusions

System Design Trade-offs

With the arrival of portable computer systems offering high level languages such as APL and significant amounts of fast main memory and large files on removeable auxiliary memories, it has become possible to implement economically, significant stand-alone testing programs. The memory sizes now available range from tens to hundreds of thousands of characters. The cost of fast, main memories now available is less than 20 cents per character while that of auxiliary storage from cartridges of magnetic tape is less than 0.01 cents per character. The availability of high-level language interpreters via read-only-memory now makes it possible to implement sophisticated systems with reduced requirements for high-speed main memory. In the implementation of this system, a reduction in the requirement of a single module of 16,384 characters of main memory can be traded for 30 million characters of auxiliary tape which is useful for the storage of teaching items and scores. This amount of auxiliary storage when used for testing items and reinforcement can accomodate 150 different tests of approximately 100 items each. With the current arrangement of 5 tests per course it is possible to provide storage for 30 courses at 100 dollars per course in place of an additional main memory module of 16,384 characters. Economies in the requirements for main memory are readily realized by implementing sophisticated CAI application with the highly structured and concise functions of APL.

Time-Variable Mastery Learning Environment

Computer delivered testing, with low-cost portable computer systems, has been demonstrated to have the advantages of permitting self-pacing and providing an ideal vehicle for organizing mastery learning courses of study. A previous paper found that students undergoing such treatment scored higher on achievement examinations, perceived that they had learned more, and accomplished significantly more out-of-class course-related activity than did a control group (Evans and Surkan, 1977). The magnetic tape storage of the item texts, combined with electronic editing from the screen of a CRT, allows continuing course refinements through additions to or deletions from the items. Also using the auxiliary storage and program control in testing makes it practical to record scores automatically, for all the randomly selected items, grouped along with complete identifying information. The completeness of the score records makes possible, subsequently, extensive analysis of individual item and student performance.

The results of such analysis serve as a source of feedback information for improving the items and for monitoring the progress of individual students. Consequently, this system has been found most satisfactory as a vehicle for testing and instruction in a mastery learning course of study. The consistency of computer testing and the practicality of the unlimited availability of a testing facility that is necessary for self-pacing has led to continued refinements in using of the system and to the extension of its application to a second course in the 1976-1977 fall semester, and a third and fourth course during 1977-1978.

References

Bieler, R. Instructor's manual to accompany Psychology applied to teaching. Houghton-Mifflin Co., Boston, 1974.

Block, J. H. Mastery learning, theory and practice. Holt, Rinehart and Winston, Inc., 1971.

Evans, R. M., and Surkan, A. J. Effect of computer-delivered testing on achievement in a mastery learning course of study with partial scoring and variable pacing. Proceedings, Eighth Conference on Computers in the Undergraduate Curricula. East Lansing: Michigan State University, 1977.

Falkoff, Adin D., and Iverson, Kenneth E. The design of APL, IBM Journal of Research and Development 17, No. 4, July 1973, pp. 324-334.